1: Difference between Samba/SMB and NFS:

Research the difference between Samba/SMB and NFS. Provide a description of the difference in at least 500 words. In addition, provide a diagram of each to add clarity to your summary. Cite your references.

     Samba/SMB and NFS are both file sharing protocols used to access files and resources over a shared network as if they were on the local device. These protocols both serve the same purpose but differ in the way they operate. While both file sharing protocols can function on any established operating system, one of the two protocols usually works better, and there are potential compatibility and configuration problems that can be run into with the less compatible protocol for the system.
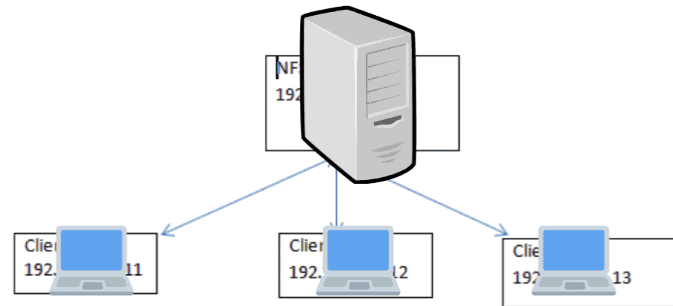
     NFS stands for Network File System and is popular on UNIX and UNIX-based operating systems, meaning it works well with Linux machines and the MAC-OS. NFS was developed by Sun Microsystems in the 1980s and uses Remote Procedure Call to communicate between client and server systems. A key disadvantage of NFS is that it does not natively support encryption meaning that the data sent over an NFS transfer could possibly be intercepted by an unwanted third party. However, other encryption tools can be used with NFS to prevent this problem. On the other hand, SMB does provide encryption natively. Interestingly, NFS performance reading encrypted data exceeds the performance of SMB.

     In contrast, Samba/SMB stands for Server Message Block and is typically used in Windows systems, being the foundation for Windows distributed file system. Since it supports the distributed file system file sharing and distribution can be done across multiple servers. Of the two file sharing protocols, SMB is typically a little easier to use in a hybrid environment, between a Windows and Linux machine, than NFS is. However, Samba, the client software for SMB which allows hybrid compatibility, is an open-source project meaning there are sometimes bugs that need to be fixed or features that need to be updated before compatibility is achieved. NFS is meant for use on UNIX and Linux but with the correct client software it can be used with other operating systems.
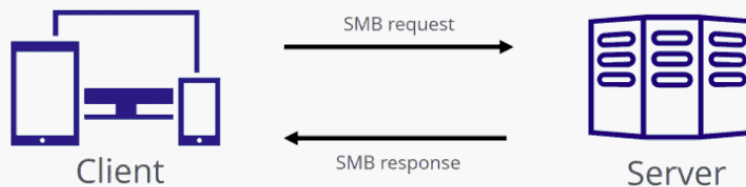
     While NFS uses the Unix command line to perform the file sharing, SMB clients mount to files and directories using an IP address or the computer hostname. SMB relies on the network port used for TCP/IP which allows for smooth file sharing even across various platforms. Windows login credentials can be used to access shared resources with SMB. Put simply, NFS uses a host-based verification system while SMB uses a user-based verification system.

     Unlike NFS that uses the RPC protocol, SMB uses a message-based protocol for client and server communication making it highly efficient and allowing it to support file locking, which prevents users from accessing the same file at the same time, along with other features. SMB makes this file locking feature mandatory whereas NFS makes file locking either mandatory or advisory depending on how it is configured. Some other features SMB includes are fast file find, and server and printer browsing which are features that NFS does not come with. In general, NFS can be more challenging to configure while SMB is relatively simple to configure, providing users with a user-friendly interface to access their shared resources. In general, NFS works better with small files, and while SMB can also work with small files, SMB typically works better with large files.

NFS Server

Server Message Block (SMB)

Referenced:
https://www.computerweekly.com/feature/NFS-vs-SMB-vs-CIFS-File-storage-protocols-defined#:~:text=NFS%20runs%20in%20Unix%2FLinux,has%20server%20and%20printer%20browsing.

https://www.writeclick.co.il/nfs-vs-smb-a-crash-course-on-network-file-sharing/

https://cloudinfrastructureservices.co.uk/nfs-vs-smb/

2: Install/configure both Samba/SMB and NFS on Linux virtual server:

Install and configure both Samba/SMB and NFS on your assigned virtual machine. Create the directories /data/smb and /data/nfs. Present these two directories as network-attached storage using the table below to determine the virtual "root" of the storage.

Service	Directory mount point
Samba/SMB	/data/smb
NFS	/data/nfs

Installing NFS on Cent-OS 7

Command: yum install nfs-utils

```
[root@lamp-07 ~]# yum install nfs-utils
Loaded plugins: fastestmirror
Determining fastest mirrors
epel/x86_64/metalink                                    |  23 kB      00:00
 * base: mirrors.seas.harvard.edu
 * epel: paducahix.mm.fcix.net
 * extras: mnvoip.mm.fcix.net

Installed:
  nfs-utils.x86_64 1:1.3.0-0.68.el7.2

Dependency Installed:
  gssproxy.x86_64 0:0.7.0-30.el7_9      keyutils.x86_64 0:1.5.8-3.el7        libbasicobjects.x86_64
  libevent.x86_64 0:2.0.21-4.el7        libini_config.x86_64 0:1.3.1-32.el7  libnfsidmap.x86_64 0:0
  libref_array.x86_64 0:0.1.5-32.el7    libtirpc.x86_64 0:0.2.4-0.16.el7     libverto-libevent.x86_
  quota-nls.noarch 1:4.01-19.el7        rpcbind.x86_64 0:0.2.0-49.el7        tcp_wrappers.x86_64 0:

Complete!
[root@lamp-07 ~]#
```

Installing SMB on Cent-OS 7

Command: yum install samba

```
[root@lamp-07 ~]#
[root@lamp-07 ~]# yum install samba
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
epel/x86_64/metalink                                    |  15 kB      00:00
 * base: centos.hivelocity.net
 * epel: epel.mirror.constant.com
 * extras: southfront.mm.fcix.net
 * remi-php80: mirror.pit.teraswitch.com
 * remi-safe: mirror.pit.teraswitch.com
 * updates: veronanetworks.mm.fcix.net
base                                                    | 3.6 kB      00:00
extras                                                  | 2.9 kB      00:00
ius                                                     | 1.3 kB      00:00
mariadb                                                 | 3.4 kB      00:00
remi-php80                                              | 3.0 kB      00:00
remi-safe                                               | 3.0 kB      00:00
updates                                                 | 2.9 kB      00:00
Package samba-4.10.16-24.el7_9.x86_64 already installed and latest version
Nothing to do
[root@lamp-07 ~]#
```

Creating directory for NFS file sharing
Command: mkdir /data/nfs

Checking if directory creation is successful: **find /data/nfs**

```
[root@lamp-07 ~]# mkdir /data/nfs
[root@lamp-07 ~]#
[root@lamp-07 ~]#
[root@lamp-07 ~]#
[root@lamp-07 ~]#
[root@lamp-07 ~]# find /data/nfs
/data/nfs
[root@lamp-07 ~]#
```

Creating directory for SMB file sharing
Command: **mkdir /data/smb**

Checking if directory creation is successful: **find /data/smb**

```
[root@lamp-07 ~]# mkdir /data/smb
[root@lamp-07 ~]#
[root@lamp-07 ~]#
[root@lamp-07 ~]#
[root@lamp-07 ~]# find /data/smb
/data/smb
[root@lamp-07 ~]#
```

*I was able to complete the assignment using the directories **/data/smb** and **/data/nfs.** I didn't run into any permission issues so I didn't change to using /tmp/nfs or /tmp/smb.

Configuring NFS on Cent-OS 7

Adding permissions and ownership rights to the directory /data/nfs
Command: **chmod –R 755 /data/nfs**
        chown nfsnobody:nfsnobody /data/nfs

```
[root@lamp-07 ~]#
[root@lamp-07 ~]# chmod -R 755 /data/nfs
[root@lamp-07 ~]# chown nfsnobody:nfsnobody /data/nfs
[root@lamp-07 ~]#
```

Enabling the NFS services
Command: **systemctl enable rpcbind**
        systemctl enable nfs-server
        systemctl enable nfs-lock
        Systemctl enable nfs-idmap

```
[root@lamp-07 ~]#
[root@lamp-07 ~]# systemctl enable rpcbind
[root@lamp-07 ~]# systemctl enable nfs-server
Created symlink from /etc/systemd/system/multi-user.target.wants/nfs-server.serv
ice to /usr/lib/systemd/system/nfs-server.service.
[root@lamp-07 ~]#
[root@lamp-07 ~]# systemctl enable nfs-lock
[root@lamp-07 ~]# systemctl enable nfs-idmap
[root@lamp-07 ~]#
```

Starting the NFS services
Command: **systemctl start rpcbind**
        systemctl start nfs-server
        systemctl start nfs-lock
        systemctl start nfs-idmap

```
[root@lamp-07 ~]# systemctl start rpcbind
[root@lamp-07 ~]# systemctl start nfs-server
[root@lamp-07 ~]# systemctl start nfs-lock
[root@lamp-07 ~]# systemctl start nfs-idmap
[root@lamp-07 ~]# █
```

Editing the configuration file for NFS
Command: **vi /etc/exports**
Line to add to file: **/data/nfs   172.16.254.29(rw,sync,no_root_squash)**

*This is the name of the directory to be shared, then the IP address of the
client you want to connect to.*

:wq! Closing and saving text editor

*Note that this file was not created upon installation of NFS, had to manually
create and add to the file. Still operates as the configuration file for NFS once
created.

```
 root@lamp-07:~
/data/nfs          172.16.254.29(rw,sync,no_root_squash)
~
~
~
```

After editing /etc/exports restart the nfs-server
```
[root@lamp-07 ~]#
[root@lamp-07 ~]# systemctl restart nfs-server
[root@lamp-07 ~]#
```

Reconfigure the firewall
Commands: **firewall-cmd --permanent --zone=public --add-service=nfs**
        firewall-cmd --permanent --zone=public --add-service=mountd
        firewall-cmd --permanent --zone=public --add-service=rpc-bind
        firewall-cmd --reload

```
[root@lamp-07 ~]#
[root@lamp-07 ~]# firewall-cmd --permanent --zone=public --add-service=nfs
success
[root@lamp-07 ~]# firewall-cmd --permanent --zone=public --add-service=mountd
success
[root@lamp-07 ~]# firewall-cmd --permanent --zone=public --add-service=rpc-bind
success
[root@lamp-07 ~]# firewall-cmd --reload
success
[root@lamp-07 ~]#
```

Configuring SMB on Cent-OS 7

Starting and enabling the SMB services
Command: systemctl start smb.service
         systemctl start nmb.service
         systemctl enable smb.service
         systemctl enable nmb.service

```
[root@lamp-07 ~]# systemctl start smb.service
[root@lamp-07 ~]# systemctl start nmb.service
[root@lamp-07 ~]# systemctl enable smb.service
Created symlink from /etc/systemd/system/multi-user.target.wants/smb.service to
/usr/lib/systemd/system/smb.service.
[root@lamp-07 ~]#
[root@lamp-07 ~]# systemctl enable nmb.service
Created symlink from /etc/systemd/system/multi-user.target.wants/nmb.service to
/usr/lib/systemd/system/nmb.service.
[root@lamp-07 ~]#
```

Configuring the firewall so that samba will work

Commands: firewall-cmd --permanent --zone=public --add-service = samba
          firewall-cmd --zone=public --add-service = samba

```
[root@lamp-07 ~]# firewall-cmd --permanent --zone=public --add-service=sam
ba
success
[root@lamp-07 ~]#
[root@lamp-07 ~]# firewall-cmd --zone=public --add-service=samba
success
[root@lamp-07 ~]#
```

Reloading the firewall after making necessary changes

Command: firewall-cmd --reload

```
[root@lamp-07 ~]# firewall-cmd --reload
success
```

Created a new user for samba share
Command: useradd sambauser -s /sbin/nologin
         smbpasswd -a sambauser
Password: HelpyouIcan2!

```
[root@lamp-07 ~]# useradd sambauser -s /sbin/nologin
[root@lamp-07 ~]# smbpasswd -a sambauser
New SMB password:
Retype new SMB password:
Added user sambauser.
[root@lamp-07 ~]#
```

Change permissions and ownership rights for /data/smb which will be the shared
directory

Commands: chown sambauser /data/smb
          chmod 755 /data/smb
```
[root@lamp-07 ~]#
[root@lamp-07 ~]# chown sambauser /data/smb
[root@lamp-07 ~]# chmod 755 /data/smb
[root@lamp-07 ~]#
```

Opened the text editor for the samba configuration file
Command: vi /etc/samba/smb.conf
```
[root@lamp-07 ~]# vi /etc/samba/smb.conf
[root@lamp-07 ~]#
```

Added the section called [sambauser] to the configuration file.
Then saved and closed the file.
[sambauser]
path=/data/smb
browseable = yes
read only = no
writeable = yes
guest ok = no
valid users = sambauser

*The text in the brackets is the name of the file share which will be important
when mounting the network share to the Windows client.*
```
        write list = @printadmin root
        force group = @printadmin
        create mask = 0664
        directory mask = 0775


[sambauser]
        path=/data/smb
        browseable = yes
        read only = no
        writeable = yes
        guest ok = no
        valid users = sambauser


```

After making and saving changes to the configuration file, restart smb

Command: systemctl restart smb nmb

```
[root@lamp-07 ~]#
[root@lamp-07 ~]# systemctl restart smb nmb
[root@lamp-07 ~]#
```

Referenced:
https://elearning.wsldp.com/pcmagazine/install-centos-samba-server/
https://dev.to/prajwalmithun/setup-nfs-server-client-in-linux-and-unix-27id

3: Mount network storage on both a Windows client and a Linux client:

Mounting network storage on a Linux client:

*Used Ubuntu laptop from assignment 5

Install nfs to Ubuntu
Command: apt install nfs-common

```
root@maloneruth-Latitude-5580:~#
root@maloneruth-Latitude-5580:~# apt install nfs-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  keyutils libevent-core-2.1-7 libnfsidmap1 rpcbind
Suggested packages:
  open-iscsi watchdog
The following NEW packages will be installed:
  keyutils libevent-core-2.1-7 libnfsidmap1 nfs-common rpcbind
0 upgraded, 5 newly installed, 0 to remove and 157 not upgraded.
Need to get 475 kB of archives.
After this operation, 1,709 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 libevent-core
Get:2 http://us.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnf
Get:3 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 rpcbind amd64
Get:4 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 keyutils amd6
Get:5 http://us.archive.ubuntu.com/ubuntu jammy-updates/main amd64 nfs-c
Fetched 475 kB in 0s (1,217 kB/s)
Selecting previously unselected package libevent-core-2.1-7:amd64.
(Reading database ... 202797 files and directories currently installed.)
```

Make directory for file share
Command: mkdir –p /mnt/data/nfs

Connect to OS class Wi-Fi

Run command to mount the network-attached storage
Command: mount 172.16.9.68:/data/nfs /mnt/data/nfs

*mount (ipaddressoftheserver):/nameofshareddirectoryonserver
nameofshareddirectoryonclient*

Confirm that the mount is successful
Command: df -kh

```
root@maloneruth-Latitude-5580:~# mount 172.16.9.68:/data/nfs /mnt/data/nfs
root@maloneruth-Latitude-5580:~#
root@maloneruth-Latitude-5580:~# df -kh
Filesystem              Size  Used Avail Use% Mounted on
tmpfs                   784M  2.2M  781M   1% /run
/dev/sda2               234G   12G  210G   6% /
tmpfs                   3.9G     0  3.9G   0% /dev/shm
tmpfs                   5.0M  4.0K  5.0M   1% /run/lock
/dev/sda1               511M  6.1M  505M   2% /boot/efi
tmpfs                   784M  116K  783M   1% /run/user/1000
172.16.9.68:/data/nfs    16G  130M   15G   1% /mnt/data/nfs
root@maloneruth-Latitude-5580:~#
```

Next, make the mount permanent by editing the /etc/fstab file

Command: nano /etc/fstab
Add line: **172.16.9.68:/data/nfs /mnt/data/nfs nfs defaults 0 0**

```
  GNU nano 6.2                                              /etc/fstab *
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name d
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>          <dump>  <pas
# / was on /dev/sda2 during installation
UUID=55ba9836-3443-44a7-ae44-71dd5d02e5e5 /                ext4    er
# /boot/efi was on /dev/sda1 during installation
UUID=6332-7DB4  /boot/efi        vfat    umask=0077         0       1
/swapfile                                none               swap    sw


172.16.9.68:/data/nfs /mnt/data/nfs nfs defaults 0 0



^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Ex
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Ju
```

Test that the NFS share works correctly

Command: echo "client hello" >> /mnt/data/nfs/testing.txt

```
root@maloneruth-Latitude-5580:~#
root@maloneruth-Latitude-5580:~# echo "Client hello" >> /mnt/data/nfs/testing.txt
root@maloneruth-Latitude-5580:~#
```

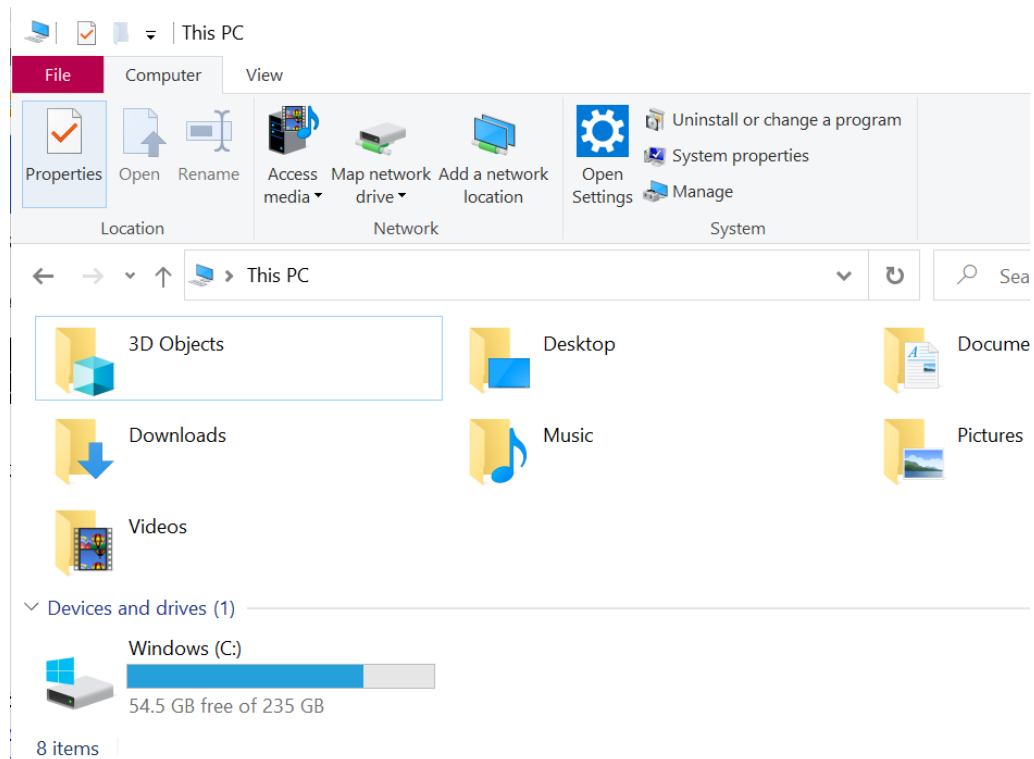Switch to server: **cat /data/nfs/testing.txt**

```
[root@lamp-07 ~]#
[root@lamp-07 ~]# cat /data/nfs/testing.txt
Client hello
[root@lamp-07 ~]#
```

Mounting network storage on a Windows client:

First open this window on your windows client and open **"Map network drive"**

Choose the drive letter you want, Z: was preselected for me.
Confirm your client is on the same Wifi network as the server.
Enter the ip address of the server, then the share name
\\172.16.9.68\sambauser
The share name is the text in brackets from the samba configuration file,
[sambauser].

Map Network Drive

What network folder would you like to map?

Specify the drive letter for the connection and the folder that you want to connect to:

Drive: Z:

Folder: \\172.16.9.68\sambauser   Browse...

Example: \\server\share

☑ Reconnect at sign-in

☐ Connect using different credentials

Connect to a Web site that you can use to store your documents and pictures.

Finish   Cancel

The first time you mount it will ask for credentials for the user, this is the username and password you configured for samba
Username: sambauser
Password: HelpyouIcan2!
Select "Remember my credentials" so you don't have to login every time you open the shared network folder.

Windows Security   ✕
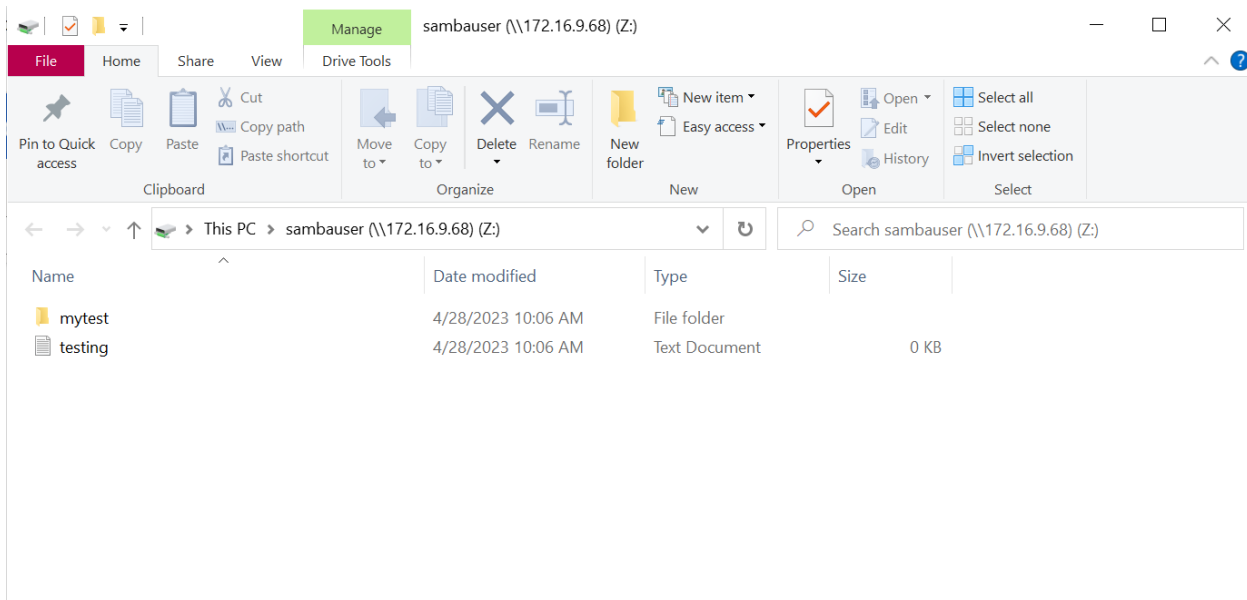
Enter network credentials

Enter your credentials to connect to: 172.16.9.68

sambauser

••••••••••••

☑ Remember my credentials

More choices

OK   Cancel

Once successfully connected, I added a folder called mytest and a document called testing on the Windows client side.

To confirm the mount for network-attached storage was successful, I ran the following commands on the server side. I changed the directory to the shared directory, /data/smb and listed the files in the directory. The files I created on the Windows client are also on the lamp server.

Commands: cd /data/smb
**   ls -l**

Referenced:
https://www.digitalocean.com/community/tutorials/how-to-set-up-an-nfs-mount-on-ubuntu-20-04

https://www.techrepublic.com/article/how-to-connect-to-linux-samba-shares-from-windows-10/